CLAIMS

In the claims:

1.   (Currently Amended) A method of queuing ~~request~~ requests to access to a server having software with a set number of available licenses, the method comprising:

receiving requests for access to the software on the server from a plurality of remote users;

allowing access to the software on the server to some of the plurality of remote users such that the number of remote users allowed access does not exceed the set number of available licenses;

sending a message to any remote user denied access, the message indicating that an access is not possible and that the user denied access will be notified when access is available.

placing remote users denied access in a queue;

sending an alert to a queued remote user in the queue when a license becomes available, the alert indicating that access is available; and

determining whether the queued remote user has responded to the alert so as to obtain access;

allowing access to the software on the server to the queued remote users but only after the queued remote user responds to the alert.

2.   (Previously Presented) The method of claim 1, further comprising placing the queued remote users denied access back in the queue if the queued remote user does not respond to the alert, to allow the queued remote users an additional opportunity to respond when an additional license becomes available.

3.   (Previously Presented) The method of claim 2, wherein each of the queued remote users is allowed only a predetermined number of additional opportunities to respond to the alert before terminating the request for access.

4.   (Original) The method of claim 1, wherein the remote users in the queue are prioritized based on when the requests are received.

5.   (Currently Amended) A server comprising:

a receiver to receive requests for access to a software on the server from a plurality of remote users, the software having a set number of available licenses;

a processor to allow access to the software on the server to some of the plurality of remote users such that the number of remote users allowed access does not exceed the set number of available licenses, to generate and send a message to any remote users denied access, the message indicating that an access is not possible and that the user denied access will be notified when access is available, and to place remote users denied access in a queue;

a transmitter to send an alert to a queued remote user in the queue when a license becomes available, the alert indicating that access is available;

wherein the processor determines whether the queued remote user has responded to the alert so as to obtain access, and allows access to the software to the queued remote users but only after the queued remote users respond to the alert.

6.   (Previously Presented) The server of claim 5, wherein the processor places the remote users denied access back in the queue if the queued remote user does not respond to the alert to allow the queued remote user an additional opportunity to respond when an additional license becomes available.

7.   (Previously Presented) The server of claim 6, further comprising a counter to count a predetermined number of returns to the queue wherein each of the queued remote users is

allowed only the predetermined number of additional opportunities to respond to the alert before terminating the request for access.

8.   (Original) The server of claim 5, wherein the remote users in the queue are prioritized based on when the requests are received.

9.   (Currently Amended) A computer-readable medium having stored thereon data representing instructions that, when executed by a processor of a server, cause the processor to perform operations comprising:

receiving requests for access to software on the server from a plurality of remote users, the software having a set number of available licenses;

allowing access to the software on the server to some of the plurality of remote users such that the number of remote users allowed access does not exceed the set number of available licenses;

sending a message to any remote users denied access, the message indicating that an access is not possible and that the user denied access will be notified when access is available.

placing the remainder of the plurality of remote users in a queue;

sending an alert to a queued remote user in the queue when a license becomes available, the alert indicating that access is available; and

determining whether the queued remote user has responded to the alert;

allowing access to the software on the server to the queued remote users but only after the queued remote users respond to the alert.

10.   (Previously Presented) The computer-readable medium of claim 9, wherein the instructions further cause the processor to place the remote users denied access back in the queue if the queued remote user does not respond to the alert, to allow the queued remote user an additional opportunity to respond when an additional license becomes available.

11. (Previously Presented) The computer-readable medium of claim 10, wherein each of the queued remote users is allowed only a predetermined number of additional opportunities to respond to the alert before terminating the request for access.

12. (Original) The computer-readable medium of claim 9, wherein the remote users in the queue are prioritized based on when the requests are received.

13. (Previously Presented) The server of claim 5 wherein one or more ports are reserved exclusively for receiving requests from remote users.

14. (Previously Presented) The server of claim 5 wherein the queued remote user must respond within a predetermined time period.

15. (Previously Presented) The server of claim 5 wherein information about the remote users is stored by the server, the server terminates contact with the queued remote user, and uses the information to contact the remote users as licenses become available.